A Joint Optimization for Dynamic Federated Learning in UAV-aided Digital Twin Vehicular Networks

2023 - CCL Research Progress Seminar

Giang Pham

May 17, 2023

Contents

Introduction

Digital Twin (DT)

- An intelligent system, digitally replicate a physical object (PO) on the cloud or MEC server [7]
- To model, analyze, predict, optimize PO during its life cycle
- Consist of 3 parts:
  - a PO (robot, car, complex system, ...)
  - virtual twin of PO
  - connection bw. PO and their twin
- 2 connection types: (i) physical-to-twin, (ii) twin-to-physical

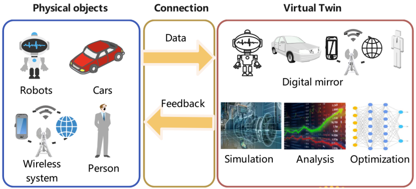


Figure 1: Digital twin concept

Digital Twin Vehicular Networks (DTVN)

- DT models of the road, traffic states, parking space, vehicle's movement status (speed, direction, ...)
- Related applications [5]:
  - Road planning according to weather, and traffic conditions
  - Smart parking in free spaces by the information the parking DT
  - Vehicle diagnosis from real-time status of autonomous vehicles
  - Personalized service recommendations as recommending food, entertainments
- Making the connection between the twins to form a DTVN

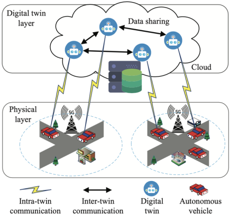


Figure 2: Digital twin vehicular network

Federated Learning (FL)

- A distributed manner to model the twin: users (UE) send locally trained model parameters instead of raw data (the centralized manner) to the server
- Compared with the centralized manner:
  - Reduce the risk of data leakage, preserve user privacy
  - Reduce communication burden
- FL is an iterative procedure: (i) UEs receive the initial model parameters w_0 from BS $\rightarrow$ *(ii)* UEs locally train the model based on it own data to get w_k $\rightarrow$ *(iii)* Users send the w_k to BS server $\rightarrow$ *(iv)* BS aggregate (average) the model w_0 $\rightarrow$ *(v)* BS broadcast w_0 to UEs

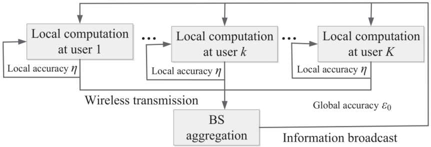


Figure 3: Federated learning procedure [8]

DTVN's Quality of Service

To guarantee the QoS of DTVN, 2 requirements:

- Archive the global accuracy ϵ_0 of the twin model
- Within the target time duration τ

Trade-off between time and energy consumption

Our work objective:

Minimize the energy consumption while satisfing the global accuracy ϵ_0 and target time requirement τ of the FL process to construct DTVN

Related Work & Our Proposal

Due to the frequent data exchange *(model parameters)* bw. UEs and BS, the communications issue can be a bottleneck in the FL process Other done work:

- [6]: cooperative relay of FL computing nodes, minimize the loss and energy consumption with time constraint
- [9]: cooperative relay energy energy efficiency with maximize amount of transmission data with minimize the energy with intensive relay
- [8]: formula i, n energy efficient with formula of rounds in static network
- [1]: formula i, n client selection to maximize selected clients while minimizing energy and satisfying time constraint
- [4]: formula i, n minimize weighted sum of time and energy, with transmission time modelled as packet delay

Our proposal:

- Dynamic network where CSI changes due to the moving of vehicles
- Deploy UAV as a relay node to eliminate the impact of the communication issue
- Dynamic update formula of i, n

System Model

System Model

Scenario: Constructing the DT vehicular network by FL

- A base station BS with integrated MEC server
- K moving VEHs on the road, high density at the intersection, the road is at the edge of BS's coverage area
- A relay node UAV, fixed hovering near the intersection

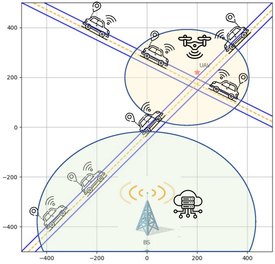


Figure 4: UAV-aided digital twin vehicular networks

Federated Learning Model

- Data:
  - Each VEH has a local dataset $\mathcal{D}_k$ with size D_k data samples
  - $\mathcal{D}_k = \{\mathbf{x}_{kl}, y_{kl}\}_{l=1}^{D_k}$, $\mathbf{x}_{kl} \in \mathbb{R}^d$ with d: dimension of input data
- FL model:
  - Local FL loss function:

$$F_k(\mathbf{w}) = \frac{1}{D_k} \sum_{l=1}^{D_k} f(\mathbf{w}, \mathbf{x}_{kl}, y_{kl}) \tag{1}$$

  - Global FL training problem - optimize the global model

$$\min_{\mathbf{w}} F(\mathbf{w}) = \sum_{k=1}^{K} \frac{D_k}{D} F_k(\mathbf{w}) = \frac{1}{D} \sum_{k=1}^{K} \sum_{l=1}^{D_k} f(\mathbf{w}, \mathbf{x}_{kl}, y_{kl}) \tag{2}$$

But each VEH has only a subset of the data, how to find the global model that generalizes well for all VEHs?

Federated Learning - Surrogate Loss Function

Adding a surrogate term to the original local loss function [1]

$$\min_{\mathbf{h}_k} G_k(\mathbf{w}^{(n)}, \mathbf{h}_k) \triangleq F_k(\mathbf{w}^{(n)} + \mathbf{h}_k) - \langle \nabla F_k(\mathbf{w}^{(n)}) - \xi \nabla F(\mathbf{w}^{(n)}), \mathbf{h}_k \rangle \quad (3)$$

(Like the form of Taylor approximation of original local loss function [2])

- ξ: weight factor of global gradient
- $\mathbf{w}^{(n)}$: optimal global model params at iteration n
- $\mathbf{w}^{(n)} + \mathbf{h}_k$: optimal local model params at iteration $n+1$

[1] A surrogate model is an engineering method used when an outcome of interest cannot be easily measured or computed, so an approximate mathematical model of the outcome is used instead. Wikipedia

[2] Taylor approximation $f(x) \approx f(a) + f^{'}(x)(x-a)$

Federated Learning Algorithm - Local Optimization

- At global round n, VEH minimizes the local loss by stochastic gradient descent (SGD) for convex loss function (SGD extensions: Adam, Adagrad, ... avoid trapped at local minima of non-convex)

$$\mathbf{h}_k^{(n),(i+1)} = \mathbf{h}_k^{(n),(i)} - \delta \nabla G_k(\mathbf{w}^{(n)}, \mathbf{h}_k^{(n),(i)}), \delta\text{: learning rate} \quad (4)$$

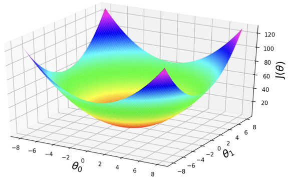


Figure 5: Convex function

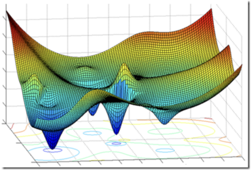


Figure 6: Non-convex function

- Convergence if reaches the local accuracy η

$$G_k(\mathbf{w}^{(n)}, \mathbf{h}_k^{(n),(i)}) - G_k(\mathbf{w}^{(n)}, \mathbf{h}_k^{(n),*}) = \eta(G_k(\mathbf{w}^{(n)}, 0) - G_k(\mathbf{w}^{(n)}, \mathbf{h}_k^{(n),*})) \quad (5)$$

Federated Learning Algorithm - Global Aggregation

- Aggregate the model parameters, global gradient:

$$\mathbf{w}^{(n)} = \mathbf{w}^{(n-1)} + \frac{1}{K}\sum_{k=1}^{K}\frac{D_k}{D}\mathbf{h}_k^{(n),*} \tag{6}$$

$$\nabla F(\mathbf{w}^{(n)}) = \frac{1}{K}\sum_{k=1}^{K}\frac{D_k}{D}\nabla F_k(\mathbf{w}^{(n)}) \tag{7}$$

- Broadcast $\mathbf{w}^{(n)}; \nabla F(\mathbf{w}^{(n)})$ to all VEHs for the next round
- Convergence if reaches the global accuracy ϵ_0

$$F(\mathbf{w}^{(n)}) - F(\mathbf{w}^{*}) = \epsilon_0(F(\mathbf{w}^{(0)}) - F(\mathbf{w}^{(*)})) \tag{8}$$

Problem Formulation

Local Training & Model Parameters Transmission

Our objective: Joint learning and communication resource allocation to minimize the energy while satisfying the DTVN's QoS

	Energy	Time
Comp.	$e_k^{\text{cp}} = \kappa C_k D_k f_k^2$	$t_k^{\text{cp}} = \frac{C_k D_k}{f_k}$
Coms.	$e_k^{\text{co}} = p_k t_k^{co}$	$t_k^{\text{co}} = \frac{s_k \ln(2)}{B \ln(1+\frac{p_k h_k}{N_0})} + x_k \delta_t$
Total	$e_k = n\big(e_k^{\text{co}} + i \times e_k^{\text{cp}}\big)$	$t_k = n\big(t_k^{\text{co}} + i \times t_k^{\text{cp}}\big)$

- i, n: # local rounds , # global rounds to reach η, ϵ_0
  $i = v \log_2(\frac{1}{\eta}), n = \frac{a}{1-\eta}$, $v = \frac{2}{(2-L\delta)\delta\gamma}, a = \frac{2L^2}{\gamma^2 \xi} \ln \frac{1}{\epsilon_0}$ [8]
  L-Lipschitz, γ-strongly convex characteristic of convex loss function
- δ_t: penalty time *if* choosing UAV
- η, f_k, p_k, x_k: optimization variables
  $x_k = 1$ *if* choosing UAV *else* 0
  $h_k = (1 - x_k) h_k^u + x_k h_k^r$

Target Latency Requirement?

- i, n: to guarantee the target global accuracy ϵ_0
  - How about the target latency: $t_k = n(t_k^{\mathrm{co}} + i \times t_k^{\mathrm{cp}}) \leq \tau$?
    t_k^{co} varies because of VEHs' movement
    $\Rightarrow$ How to guarantee? (CSI changes in each global round)
- Our idea: Solve the optimization problem *at the beginning of each round instead of the first round*
  - if the bad network condition (long t_k^{co}), we increase the local computation (but also increase e_k^{cp})
  - if the good network condition (short t_k^{co}), we decrease the local computation (and also decrease e_k^{cp})

However: n is to meet the target ϵ_0 of the whole FL

How to derive n in our idea?

Dynamic Global Accuracy Update

For the whole FL process with the desired accuracy ϵ_0:

$$F(\mathbf{w}^n) - F(\mathbf{w}^*) = \epsilon_0(F(\mathbf{w}^0) - F(\mathbf{w}^*)) \qquad \textit{(simplified!)}$$

and at each global round:

$$\begin{aligned}
0 : F(\mathbf{w}^1) - F(\mathbf{w}^*) &= \epsilon_1(F(\mathbf{w}^0) - F(\mathbf{w}^*)) \\
1 : F(\mathbf{w}^2) - F(\mathbf{w}^*) &= \epsilon_2(F(\mathbf{w}^1) - F(\mathbf{w}^*)), \dots \\
n-2 : F(\mathbf{w}^{n-1}) - F(\mathbf{w}^*) &= \epsilon_{n-1}(F(\mathbf{w}^{n-2}) - F(\mathbf{w}^*)), \\
n-3 : F(\mathbf{w}^n) - F(\mathbf{w}^*) &= \epsilon_n(F(\mathbf{w}^{n-1}) - F(\mathbf{w}^*))
\end{aligned}$$

$$\Rightarrow F(\mathbf{w}^n) - F(\mathbf{w}^*) = \epsilon_n \epsilon_{n-1} \dots \epsilon_2 \epsilon_1 (F(\mathbf{w}^{n-1}) - F(\mathbf{w}^*))$$

(Mathematical induction)

$$\Rightarrow \boxed{\epsilon_0 = \epsilon_n \epsilon_{n-1} \dots \epsilon_2 \epsilon_1}$$

Problem Formulation

We solve the optimization at the beginning of each global round [3]:

$$
\begin{aligned}
\min_{\eta, \{f_k, x_k, p_k\}_{k=1}^{K}} \quad & \sum_{k=0}^{K-1} n\left(e_k^{\mathrm{co}} + i \times e_k^{\mathrm{cp}}\right) \\
\text{s.t.} \quad & n\left(t_k^{\mathrm{co}} + i \times t_k^{\mathrm{cp}}\right) \le \tau, \\
& 0 \le \eta \le 1, \\
& 0 \le f_k \le f_k^{max}, \forall k, \\
& x_k = \{0, 1\}, \forall k, \\
& \sum_k x_k = N_0, \\
& 0 \le p_k \le p_k^{max}, \forall k
\end{aligned}
$$

in which, $n = \frac{a}{1-\eta}$, $a = \frac{2L^2}{\gamma^2 \xi} \ln \frac{1}{\epsilon_{(n)}}$; $\tau = \tau - t_{(n-1)}$(remaining time), and $\epsilon_0 = \epsilon_{(1)} \dots \epsilon_{(n-1)} \epsilon_{(n)}$

[3]For simplicity, we drop the superscript (n), which means at global round n

Choosing Dynamic Global Accuracy Update Model

How to choose $\epsilon_{(0)}, \epsilon_{(1)}, \ldots, \epsilon_{(n-1)}, \epsilon_{(n)}$? *(temp. global accuracy)*

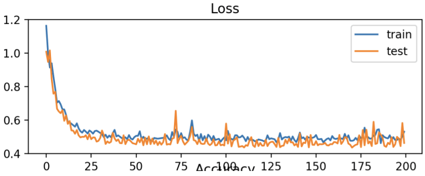


Figure 7: Landscape of loss function

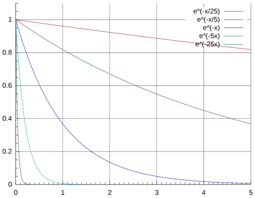


Figure 8: Exponential decay

$\Rightarrow$ We can model the dynamic behavior as decay rate:

$\epsilon_{(1)} = \alpha^0 \epsilon_{(0)}, \epsilon_{(2)} = \alpha^1 \epsilon_{(0)}, \ldots, \epsilon_{(n-1)} = \alpha^{n-2} \epsilon_{(0)}, \epsilon_{(n)} = \alpha^{n-1} \epsilon_{(0)}$

$$\boxed{\epsilon_{(1)} \ldots \epsilon_{(n-1)} \epsilon_{(n)} = \alpha^{\frac{(n-1)n}{2}} (\epsilon_{(0)})^n = \epsilon_0, \alpha > 1}$$

, $\Rightarrow$ We can choose the suitable $\epsilon_{(0)}, \alpha$

Network Optimization

Problem decomposition

We have a joint learning and communication resource allocation:

- Fixed (f_k^*, x_k^*, p_k^*), optimize η - Learning optimization (LO) ($A_k = vC_kD_k$):

$$\min_{\eta} \quad \frac{a}{1-\eta}\left(\sum_k e_k^{co} + \sum_k \frac{\kappa A_k f_k^2}{\ln 2}\ln(1/\eta)\right) \tag{9a}$$

$$\text{s.t.} \quad T_k = \frac{a}{1-\eta}\left(t_k^{co} + \frac{A_k}{f_k \ln 2}\ln(1/\eta)\right) \leq \tau, \forall k, \tag{9b}$$

$$0 \leq \eta \leq 1 \tag{9c}$$

- Fixed η^*, optimize (f_k, x_k, p_k) - Resource allocation (RA):

$$\min_{\{f_k, x_k, p_k\}_{k=1}^K} \quad \sum_k \left(p_k\left[\frac{(\ln 2)s_n/B}{\ln\left(1+\frac{p_k h_k}{N_0}\right)} + x_k\delta_t\right] + \kappa A_k \log_2(1/\eta) f_k^2\right) \tag{10a}$$

$$\text{s.t.} \quad \left(\frac{(\ln 2)s_n/B}{\ln\left(1+\frac{p_k h_k}{N_0}\right)} + x_k\delta_t\right) + A_k \log_2(1/\eta)\frac{1}{f_k} \leq \frac{\tau}{n}, \forall k, \tag{10b}$$

$$0 \leq f_k \leq f_k^{\max}, 0 \leq p_k \leq p_k^{\max}, \tag{10c}$$

$$x_k = \{0, 1\}, \sum_k x_k \leq N_0 \tag{10d}$$

We iteratively solve these 2 subproblems until convergence.

Learning Optimization

Denote $a_f^e = a \sum_k \frac{\kappa A_k f_k^2}{\ln 2}$, $b_f^e = a \sum_k e_k^{co}$, $a_f^t = a t_k^{co}$, $b_f^t = a \frac{A_k}{f_k \ln 2}$, we rewrite LO:

$$\min_{\eta} \quad \frac{b_f^e + a_f^e \ln(1/\eta)}{1-\eta} \tag{11a}$$

$$\text{s.t.} \quad T_k = \frac{b_f^t + a_f^t \ln(1/\eta)}{1-\eta} \le \tau, \forall k, \tag{11b}$$

$$0 \le \eta \le 1 \tag{11c}$$

Both 11a, 11b are in the same form, convex $\rightarrow$ solve iteratively in 2 steps:

- S1: Bound tightening of 11b by solving Larmbert-W [4] of $T_k = \tau$
  $\eta^{min} = \max_k W_0(z_k)$, $\eta^{max} = \min_k W_{-1}(z_k)$
  with $z_k = -\frac{tau}{a_f^t} \exp(\frac{b_f^t - tau}{a_f^t})$
- S2: Convex function 11a, which has a fractional form
  We use Dinkelbach method

[4] Larmbert-W function: $we^w = z$ holds iff $w = W_k(z)$, k: branch number

Resource Allocation - Frequency & Power Optimization

With fixed x_k^*, RA is a frequency and power optimization (FPO) written as

$$\min_{\{f_k, p_k\}_{k=1}^K} \quad \sum_k \left(p_k \left[\frac{(\ln 2)^{s_n}/B}{\ln\left(1 + \frac{p_k h_k}{N_0}\right)} + \Delta_t \right] + \kappa i C_n D_n f_k^2 \right) \tag{12a}$$

$$\text{s.t.} \quad \left(\frac{(\ln 2)^{s_n}/B}{\ln\left(1 + \frac{p_k h_k}{N_0}\right)} + \Delta_t \right) + i C_n D_n \frac{1}{f_k} \leq \frac{\tau}{n}, \forall k, \tag{12b}$$

$$0 \leq f_k \leq f_k^{\max}, 0 \leq p_k \leq p_k^{\max} \tag{12c}$$

with $\Delta_t = x_k \delta_t$. Denote $a = {}^{\ln(2) s_n}/_{B}$, $b = {}^{h_k}/_{N_0}$, $c = i C_n D_n$, $\tau^{'} = \tau / n - \Delta t$, substitute $z = {}^{1}/_{\ln(1+bp_k)}$, $t = 1/f_k$, we transform FPO for each k as

$$\min_{f_k, p_k} \quad \frac{a}{b} \left(\exp\left({}^{1}/_{z} - 1\right) z + \frac{\kappa c}{t^2} \right) \tag{13a}$$

$$\text{s.t.} \quad az + ct = \tau^{'}, \tag{13b}$$

$$z \geq z_{\min}, t \geq t_{\min} \tag{13c}$$

with $z_{\min} = {}^{1}/_{\ln(1+bp_k^{\max})}$, $t = 1/f_k^{\max}$. This is a linear constrained convex optimization. We solve by primal-dual interior-point method [2] with customized normalization.

Resource Allocation - Relay Selection

With fixed (f_k^*, p_k^*), relay node selection (RS) is an integer optimization.

- Step 1: $\{x_k = 1\}_k$, solve FPO $\rightarrow$ get $\{f_k^{*,\text{uav}}, p_k^{*,\text{uav}}\}_k$, $e_k^{*,\text{uav}}$
- Step 2: $\{x_k = 0\}_k$, solve FPO $\rightarrow$ get $\{f_k^{*,\text{bs}}, p_k^{*,\text{bs}}\}_k$, $e_k^{*,\text{bs}}$
- Step 3: Select x_k that gives smaller e_k while satisfying $\sum_k x_k \leq N_0$

We solve RS iteratively until convergence

Simulation Results

Simulation Settings

- Dataset MNIST: a handwritten dataset including numbers 0 - 9 [3]
  - Subsample MNIST & distribute it to each VEHs to simulate the heterogeneous network (niid data). Each VEH have only 3 labels (a part of the data), i.e, user 0 (number 0, 1, 2), user 1 (1, 2, 3), ...
  - Take 80% of # samples as training set & 20% for testing set.
    $train\_data[\#samples] = [138, 67, 109, 185, 91, 94, 73, 107, 76, 220], sum = 1160$
    $test\_data[\#samples] = [35, 17, 28, 47, 23, 24, 19, 27, 19, 55], sum = 294$

- Network parameters:

$K = 10$	$(x_{\text{uav}}, y_{\text{uav}}, z_{\text{uav}}) = (200, 220, 100)m$	$\epsilon_0 = 1e{-}3$
$N_0 = 5$	$(x_{\text{bs}}, y_{\text{bs}}, z_{\text{bs}}) = (0, -500, 0)m$	$\xi = 1$
$\delta_t = 0$	$(de_r, de_u) = (2.9, 2.3)$ path loss exponent of bs, uav	$L = 5$
$p_k^{\max} = 0.1W$	$C_n = 1.5 * 1e4$	$\gamma = 3$
$f_k^{\max} = 2GHz$	$\kappa = 1e{-}28$	$s_n = 0.3Mb$

FL Performance

We consider 4 scenarios:

- bs-fixedi: (1): bs, fixed # local rounds *i*
- bs-dyni: (2): bs, dynamic # local rounds *i*
- bs-uav-fixedi: (3): bs, uav, fixed # local rounds *i*
- bs-uav-dyni: (4): bs, uav, dyn # local rounds *i* (*our proposal!*)

Results:

- All converged at train accuracy 85.7%, train loss 0.3, test loss 0.5
- Accuracy of (4) gradually approaches (1), (2), (3)

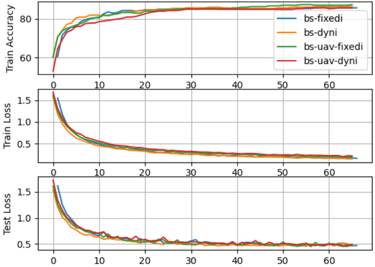
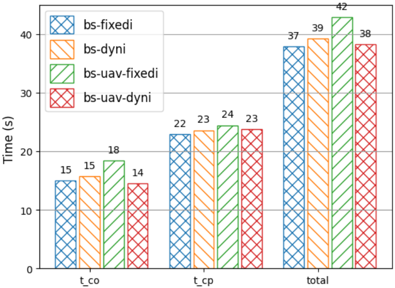
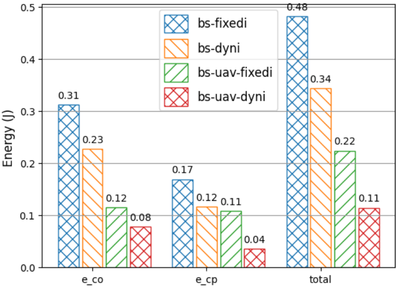


Figure 9: Convergence of FL

Network Optimization Performance





Results:

- With $\tau = 40s$, (4) give smallest energy within the required target time.

Classification Results Visualization

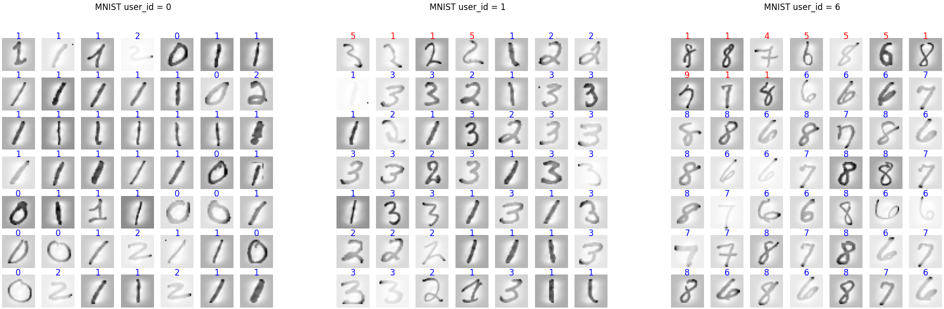


Figure 10: Classification results denote as the above numbers, in red: incorrectly classified data (of VEHs $k = 0, 1, 6$)

Results:

- Each VEH has only a subset of data, but FL generalizes well for all VEHs.

Classification Results Visualization (cont.)

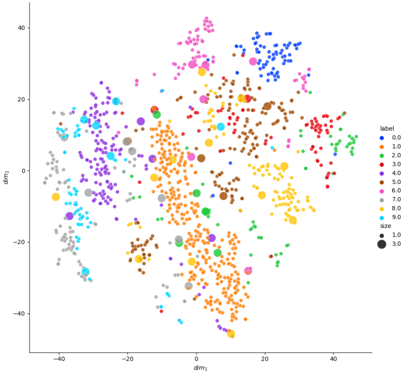


Figure 11: Label of data

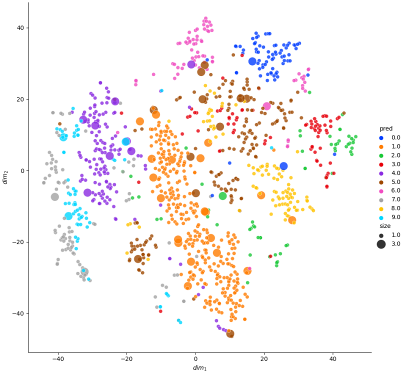


Figure 12: Classified results

Figure 13: Visualize the classification result by t-SNE method [6],
size = 3: incorrect classified data

[5] t-SNE method: a dimensionality reduction method to visualize high-dimensinal data

Unfinished Work

Unfinished Work

- Showing the impact of choosing decay value, is there any other decay form?
- Appropriate value δ_t

References

[1] Rana Albelaihi et al. "Green Federated Learning via Energy-Aware Client Selection". In: *GLOBECOM 2022 - 2022 IEEE Global Communications Conference*. 2022, pp. 13–18. DOI: 10.1109/GLOBECOM48099.2022.10001569.

[2] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[3] Li Deng. "The mnist database of handwritten digit images for machine learning research". In: *IEEE Signal Processing Magazine* 29.6 (2012), pp. 141–142.

[4] Yulan Gao et al. "Multi-Resource Allocation for On-Device Distributed Federated Learning Systems". In: *GLOBECOM 2022-2022 IEEE Global Communications Conference*. IEEE. 2022, pp. 160–165.

[5] Chao He et al. "Security and Privacy in Vehicular Digital Twin Networks: Challenges and Solutions". In: *IEEE Wireless Communications* (2022).

[6] Peichun Li et al. "FedRelay: Federated Relay Learning for 6G Mobile Edge Intelligence". In: *IEEE Transactions on Vehicular Technology* 72.4 (2023), pp. 5125–5138. DOI: 10.1109/TVT.2022.3225087.

[7] Fengxiao Tang et al. "Survey on digital twin edge networks (DITEN) toward 6G". In: *IEEE Open Journal of the Communications Society* 3 (2022), pp. 1360–1381.

[8] Zhaohui Yang et al. "Energy efficient federated learning over wireless communication networks". In: *IEEE Transactions on Wireless Communications* 20.3 (2020), pp. 1935–1949.

[9] Xinyue Zhang et al. "Energy Efficient Federated Learning over Cooperative Relay-Assisted Wireless Networks". In: *GLOBECOM 2022-2022 IEEE Global Communications Conference*. IEEE. 2022, pp. 179–184.

Thank you for your attention.